

In re Application of: John Kemeny
Attorney Docket No.: EMR-002.01

AMENDMENTS TO THE CLAIMS

1. (Currently amended) A method of data storage address translation, the method comprising:
receiving a first address in a first address space;
traversing a trie ~~based on~~ using different portions of the first address; and
determining a second address based on the traversal.
2. (Original) The method of claim 1, wherein the first address has a different address space than the second address.
3. (Original) The method of claim 2, wherein the first address has a larger address space than the second address.
4. (Original) The method of claim 1, wherein the trie includes at least one leaf identifying an address in the second address space.
5. (Original) The method of claim 1, wherein the second address comprises an address of a cache memory.
6. (Original) The method of claim 5, further comprising, based on the traversal, determining whether the cache stores information identified by the first address.
7. (Currently amended) The method of claim 6 1, wherein the trie comprises a multi-dimensional array, wherein an index of a dimension of the array corresponds to different trie branches.
8. (Currently amended) The method of claim 7, wherein traversing the trie comprises, repeatedly, indexing into the dimension of the array using a portion said different portions of the first address.

In re Application of: John Kemeny
Attorney Docket No.: EMR-002.01

9. (Original) The method of claim 5, wherein the first address comprises an address of permanent data storage.
10. (Original) The method of claim 1, wherein traversing the trie based on the first address comprises
 - performing an operation on the first address; and
 - traversing the trie using the operation results.
11. (Original) The method of claim 1, wherein the second address associated with the first address dynamically changes.
12. (Currently amended) A data storage system, comprising:
 - (a) a storage area having a first address space;
 - (b) a cache having a second address space; and
 - (c) instructions for causing a processor to
 - (1) receive a first address in the first address space;
 - (2) traverse a trie ~~based on~~ using different portions of the first address; and
 - (3) determine a second address in the second address space based on the traversal.
13. (Original) The data storage system of claim 12, wherein the instructions further comprise instructions for causing the processor to determine whether the cache stores a block in the storage area based on the trie traversal.
14. (Original) The data storage system of claim 12, wherein the instructions for causing the processor to receive a first address comprise instructions for causing the processor to receive a first address included in a data access request received from a host connected to the data storage system.
15. (Original) The data storage system of claim 12, wherein the instructions for causing the processor to traverse the trie based on the first address comprise instructions for causing the processor to:

In re Application of: John Kemeny
Attorney Docket No.: EMR-002.01

perform an operation on the first address; and
traverse the trie using the operation results.

16. (Original) The data storage system of claim 12, wherein the second address associated with the first address dynamically changes.

17. (Currently amended) A computer program product, disposed on a computer readable medium, for data storage address translation, the computer program including instructions for causing a processor to:

receive a first address within a first address space;
traverse a trie based on using different portions of the first address; and
determine a second address based on the traversal.

18. (Original) The computer program of claim 17, wherein the first address has a different address space than the second address.

19. (Original) The computer program of claim 18, wherein the first address has a larger address space than the second address.

20. (Original) The computer program of claim 17, wherein the trie includes at least one branch identifying an address in the second address space.

21. (Original) The computer program of claim 17, wherein the second address comprises an address of a cache memory.

22. (Original) The computer program of claim 21, further comprising, instructions for causing the processor to, based on the traversal, determine whether the cache stores information identified by the first address.

In re Application of: John Kemeny
Attorney Docket No.: EMR-002.01

23. (Original) The computer program of claim 17, wherein the trie comprises a multi-dimensional array, wherein an index of a dimension of the array corresponds to different trie branches.

24. (Original) The computer program of claim 17, wherein the first address comprises an address of permanent data storage.

25. (Original) The computer program of claim 17, wherein the instructions for causing the processor to traverse the trie comprise instructions for causing the processor to:

perform an operation on the first address; and
traverse the trie using the operation results.

26. (Original) The computer program of claim 17, wherein the second address associated with the first address dynamically changes.

27. (Currently amended) A method of data storage address translation at a system having a storage area composed of different physical devices, a shared cache for caching blocks of data in the storage area, and connections to different host processors, the method comprising:

receiving a storage area address within a storage area address space based on a request received from one of the host processors;
traversing a trie ~~based on~~ using different portions of the storage area address, the traversing identifying a trie leaf identifying a cache address in a cache address space; and
changing the cache address associated with the trie leaf based on system alteration of cache contents.

28. (Currently amended) A memory for storing data for access by an application program being executed on a data processing system, comprising a data structure stored in said memory, said data structure including information corresponding to a trie, the trie having leaves identifying different respective cache addresses, where each of the leaves is accessed by traversing the trie using different portions of a corresponding storage area address.

In re Application of: John Kemeny
Attorney Docket No.: EMR-002.01

29. (Canceled)

30. (Original) The memory of claim 28, wherein the trie comprises a multi-dimensional array.

31. (Previously presented) A method comprising:

representing a plurality of memory addresses associated with at least one storage device as a plurality of hierarchical branches in a trie data structure;

receiving a request for data stored at a particular one of the memory addresses of the at least one storage device;

traversing at least some of the hierarchical branches in the trie data structure using different portions of the particular one of the memory addresses storing the requested data;

identifying a memory address associated with a cache memory based on the traversal, the identified cache memory address being stored in a leaf of the trie data structure; and

transmitting data stored at the identified cache memory address in response to the request.

32. (New) The method of claim 31, wherein the trie data structure comprises a multi-dimensional array, wherein an index of a dimension of the array corresponds to different trie data structure branches.

33. (New) The method of claim 32, wherein traversing the trie data structure comprises, repeatedly, indexing into the dimension of the array using the different portions of the particular one of the memory addresses.

34. (New) The method of claim 31, wherein traversing the trie data structure using the different portions of the particular one of the memory addresses comprises

performing an operation on at least one of the different portions of the particular one of the memory addresses; and

traversing the trie data structure using the operation results.

35. (New) The method of claim 31, wherein the cache memory address associated with the particular one of the memory addresses dynamically changes.

In re Application of: John Kemeny
Attorney Docket No.: EMR-002.01

36. (New) A data storage system, comprising:

(a) at least one storage device;

(b) a cache memory;

(c) instructions for causing a processor to:

(1) represent a plurality of memory addresses associated with the at least one storage device as a plurality of hierarchical branches in a trie data structure;

(2) receive a request for data stored at a particular one of the memory addresses of the at least one storage device;

(3) traverse at least some of the hierarchical branches in the trie data structure using different portions of the particular one of the memory addresses storing the requested data;

(4) identify a memory address associated with the cache memory based on the traversal, the identified cache memory address being stored in a leaf of the trie data structure; and

(5) transmit data stored at the identified cache memory address in response to the request.

37. (New) The data storage system of claim 36, wherein the instructions for causing the processor to traverse the trie data structure based on the different portions of the particular one of the memory addresses comprise instructions for causing the processor to:

perform an operation on at least one of the different portions of the particular one of the memory addresses the first address; and

traverse the trie data structure using the operation results.

38. (New) The data storage system of claim 36, wherein the cache memory address associated with the particular one of the memory addresses dynamically changes.

39. (New) A computer program product, disposed on a computer readable medium, the computer program product including instructions for causing a processor to:

represent a plurality of memory addresses associated with at least one storage device as a plurality of hierarchical branches in a trie data structure;

In re Application of: John Kemeny
Attorney Docket No.: EMR-002.01

receive a request for data stored at a particular one of the memory addresses of the at least one storage device;

traverse at least some of the hierarchical branches in the trie data structure using different portions of the particular one of the memory addresses storing the requested data;

identify a memory address associated with a cache memory based on the traversal, the identified cache memory address being stored in a leaf of the trie data structure; and

transmit data stored at the identified cache memory address in response to the request.

40. (New) The computer program product of method of claim 39, wherein the trie data structure comprises a multi-dimensional array, wherein an index of a dimension of the array corresponds to different trie data structure branches.

41. (New) The computer program product of claim 40, wherein the instructions for causing the processor to traverse the trie data structure comprise instructions for causing the processor to, repeatedly, index into the dimension of the array using the different portions of the particular one of the memory addresses.

42. (New) The computer program product of claim 39, wherein the instructions for causing the processor to traverse the trie data structure using the different portions of the particular one of the memory addresses comprises instructions for causing the processor to:

perform an operation on at least one of the different portions of the particular one of the memory addresses; and

traverse the trie data structure using the operation results.

43. (New) The computer program product of claim 39, wherein the cache memory address associated with the particular one of the memory addresses dynamically changes.